AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application:

LISTING OF CLAIMS:

1. (Currently Amended) A system for executing computing tasks in a preboot execution environment, comprising:

a computer readable medium comprising:

a language agent with a preboot execution language interpreter, the language agent from a second system; and

at least one specification for performing at least one computing task in the preboot execution environment, the at least one specification from the second system,

wherein the language agent is configured to interpret the at least one specification for performing at least one computing task in the preboot execution environment, and configured to perform the at least one computing task specified, ~~and~~

wherein the at least one specification from the second system is an encapsulation, encapsulating parameters to be resolved at the system at execution time by the preboot execution language interpreter from the second system, and

wherein the language agent is configured to generate one or more encapsulations and to interpret one or more encapsulations.

2. (Original) The system of claim 1, wherein the preboot execution language interpreter is an object-oriented language interpreter.

3. (Cancelled).

4. (Cancelled).

5. (Previously Presented) The system of claim 1, wherein the encapsulated parameters are parametric behaviors as well as parametric data.

6. (Currently Amended) A system for image installation in a preboot execution environment, comprising:

a computer readable medium comprising:

a language agent with a preboot execution language interpreter, the language agent from a second system; and

at least one specification for performing at least one task for image installation in the preboot execution environment, the at least one specification from the second system,

wherein the language agent is configured to interpret the at least one specification, and configured to perform the at least one task specified, ~~and~~

wherein the at least one specification is an encapsulation, encapsulating parameters to be resolved at the system at execution time by the preboot execution language interpreter from the second system, and

wherein the language agent is configured to generate one or more encapsulations and to interpret one or more encapsulations.

7. (Original) The system of claim 6, wherein the preboot execution language interpreter is an object-oriented language interpreter.

8. (Cancelled).

9. (Cancelled).

10. (Previously Presented) The system of claim 6, wherein the encapsulated parameters are parametric behaviors as well as parametric data.

11. (Previously Presented) The system of claim 6, wherein an image set for image installation itself is a self-describing encapsulation, containing the at least one specification as an

encapsulation which encapsulates parameters resolved by the preboot execution language interpreter at execution time.

12. (Currently Amended) A system for remote imaging in a preboot execution environment, comprising:

a computer readable medium comprising:

a language agent with a preboot execution language interpreter, the language agent from a second system; and

at least one specification for performing at least one task for remote imaging over a network in the preboot execution environment, the at least one specification from the second system,

wherein the language agent is configured to interpret the at least one specification, and configured to perform the at least one task specified, ~~and~~

wherein the at least one specification is an encapsulation, encapsulating parameters to be resolved at the system at execution time by the preboot execution language interpreter from the second system, and

wherein the language agent is configured to generate one or more encapsulations and to interpret one or more encapsulations.

13. (Original) The system of claim 12, wherein the preboot execution language interpreter is an object-oriented language interpreter.

14. (Cancelled).

15. (Cancelled).

16. (Previously Presented) The system of claim 12, wherein the encapsulated parameters are parametric behaviors as well as parametric data.

17. (Previously Presented) The system of claim 12, wherein an image set for remote imaging itself is a self-describing encapsulation, containing the at least one specification as an

encapsulation which encapsulates parameters resolved by the preboot execution language interpreter at execution time.

18. (Currently Amended) A system for remote booting over a network, comprising:

a computer readable medium comprising:

a language agent with a preboot execution language interpreter, the language agent from a second system; and

at least one specification for performing at least one task in a preboot execution environment for remotely booting a computer over a network, the at least one specification from the second system,

wherein the language agent is configured to interpret the at least one specification, and configured to perform the at least one task specified, ~~and~~

wherein the at least one specification is an encapsulation, encapsulating parameters to be resolved at the system at execution time by the preboot execution language interpreter from the second system, and

wherein the language agent is configured to generate one or more encapsulations and to interpret one or more encapsulations.

19. (Original) The system of claim 18, wherein the preboot execution language interpreter is an object-oriented language interpreter.

20. (Cancelled).

21. (Cancelled).

22. (Previously Presented) The system of claim 18, wherein the encapsulated parameters are parametric behaviors as well as parametric data.

23. (Previously Presented) The system of claim 18, wherein an image set for remote booting itself is a self-describing encapsulation, containing the at least one specification as an encapsulation which encapsulates parameters resolved by the preboot execution language interpreter at execution time.

24. (Currently Amended) A method for executing computing tasks in a preboot execution environment, comprising ~~the steps of~~:

receiving by a first system from a second system a language agent with a preboot execution language interpreter;

generating one or more encapsulations by the language agent;

interpreting one or more encapsulations by the language agent;

receiving by the first system from the second system at least one specification for performing at least one computing task in the preboot execution environment;

interpreting by the language agent the at least one specification for performing at least one computing task in the preboot execution environment; and

performing the at least one computing task specified,

wherein the at least one specification is an encapsulation, encapsulating parameters resolved at the first system at execution time by the preboot execution language interpreter from the second system.

25. (Original) The method of claim 24, wherein the preboot execution language interpreter is an object-oriented language interpreter.

26. (Cancelled).

27. (Previously Presented) The method of claim 24, wherein the encapsulated parameters are parametric behaviors as well as parametric data.

28. (Currently Amended) A method for image installation in a preboot execution environment, comprising ~~the steps of~~:

receiving by a first system from a second system a language agent with a preboot execution language interpreter;

generating one or more encapsulations by the language agent;

interpreting one or more encapsulations by the language agent;

receiving by the first system from the second system at least one specification for performing at least one task for image installation in the preboot execution environment;

interpreting by the language agent the at least one specification for performing at least one task for image installation in the preboot execution environment; and

performing the at least one task for image installation specified,

wherein the at least one specification is an encapsulation, encapsulating parameters resolved at the first system at execution time by the preboot execution language interpreter from the second system.

29. (Original) The method of claim 28, wherein the preboot execution language interpreter is an object-oriented language interpreter.

30. (Cancelled).

31. (Previously Presented) The method of claim 28, wherein the encapsulated parameters are parametric behaviors as well as parametric data.

32. (Previously Presented) The method of claim 28, wherein an image set for image installation itself is a self-describing encapsulation, containing the at least one specification as an encapsulation which encapsulates parameters resolved by the preboot execution language interpreter at execution time.

33. (Currently Amended) A method for remote imaging in a preboot execution environment, comprising ~~the steps of~~:

receiving by a first system from a second system a language agent with a preboot execution language interpreter;

generating one or more encapsulations by the language agent;

interpreting one or more encapsulations by the language agent;

receiving by the first system from the second system at least one specification for performing at least one task for remote imaging in the preboot execution environment;

interpreting by the language agent the at least one specification for performing at least one task for remote imaging in the preboot execution environment; and

performing the at least one task for remote imaging specified,

wherein the at least one specification is an encapsulation, which encapsulates parameters resolved at the first system at execution time by the preboot execution language interpreter from the second system.

34. (Original) The method of claim 33, wherein the preboot execution language interpreter is an object-oriented language interpreter.

35. (Cancelled).

36. (Previously Presented) The method of claim 33, wherein the encapsulated parameters are parametric behaviors as well as parametric data.

37. (Previously Presented) The method of claim 33, wherein an image set for remote imaging itself is a self-describing encapsulation, containing the at least one specification as an encapsulation which encapsulates parameters resolved by the preboot execution language interpreter at execution time.

38. (Currently Amended) A method for remote booting in a preboot execution environment, comprising ~~the steps of~~:

receiving by a first system from a second system a language agent with a preboot execution language interpreter;

generating one or more encapsulations by the language agent;

interpreting one or more encapsulations by the language agent;

receiving by the first system from the second system at least one specification for performing at least one task for remote booting in the preboot execution environment;

interpreting by the language agent the at least one specification for performing at least one task for remote booting in the preboot execution environment; and

performing the at least one task for remote booting specified,

wherein the at least one specification is an encapsulation, which encapsulates parameters resolved at the first system at execution time by the preboot execution language interpreter from the second system.

39. (Original) The method of claim 38, wherein the preboot execution language interpreter is an object-oriented language interpreter.

40. (Cancelled).

41. (Previously Presented) The method of claim 38, wherein the encapsulated parameters are parametric behaviors as well as parametric data.

42. (Previously Presented) The method of claim 38, wherein an image set for remote booting itself is a self-describing encapsulation, containing the at least one specification as an encapsulation which encapsulates parameters resolved by the preboot execution language interpreter at execution time.

43. (Currently Amended) A system for specifying computing tasks in a preboot execution environment, comprising:

a computer readable medium comprising:

a language agent with a preboot execution specification generator; and

a definition for at least one specification for performing at least one computing task in a preboot execution environment,

wherein the at least one specification is an encapsulation, encapsulating parameters to be resolved at a second system at execution time by a preboot execution language interpreter located at the second system, and

wherein the language agent is configured to generate one or more encapsulations and to interpret one or more encapsulations.

44. (Previously Presented) The system of claim 43, wherein the at least one specification is generated from the definition by the language agent with a preboot execution specification generator.

45. (Original) The system of claim 43, wherein the preboot execution specification generator is an object-oriented language code generator.

46. (Cancelled).

47. (Previously Presented) The system of claim 43, wherein the encapsulated parameters are parametric behaviors as well as parametric data.

48. (Currently Amended) A system for specifying tasks for image installation in a preboot execution environment, comprising:

a computer readable medium comprising:

a language agent with a preboot execution specification generator; and

a definition for at least one specification for performing at least one task for image installation in a preboot execution environment,

wherein the at least one specification is an encapsulation, encapsulating parameters to be resolved at a second system at execution time by a preboot execution language interpreter located at the second system, and

wherein the language agent is configured to generate one or more encapsulations and to interpret one or more encapsulations.

49. (Previously Presented) The system of claim 48, wherein the at least one specification is generated from the definition by the language agent with a preboot execution specification generator.

50. (Original) The system of claim 48, wherein the preboot execution specification generator is an object-oriented language code generator.

51. (Cancelled).

52. (Previously Presented) The system of claim 48, wherein the encapsulated parameters are parametric behaviors as well as parametric data.

53. (Previously Presented) The system of claim 48, wherein the at least one specification which is an encapsulation is a part of an image set for image installation, which

henceforth renders the image set itself to be a self-describing encapsulation, encapsulating parameters resolved at execution time.

54. (Currently Amended) A system for specifying remote imaging tasks in a preboot execution environment, comprising:

a computer readable medium comprising:

a language agent with a preboot execution specification generator; and

a definition for at least one specification for performing at least one task for remote imaging in a preboot execution environment,

wherein the at least one specification is an encapsulation, encapsulating parameters to be resolved at a second system at execution time by a preboot execution language interpreter located at the second system, and

wherein the language agent is configured to generate one or more encapsulations and to interpret one or more encapsulations.

55. (Previously Presented) The system of claim 54, wherein the at least one specification is generated from the definition by the language agent with a preboot execution specification generator.

56. (Original) The system of claim 54, wherein the preboot execution specification generator is an object-oriented language code generator.

57. (Cancelled).

58. (Previously Presented) The system of claim 54, wherein the encapsulated parameters are parametric behaviors as well as parametric data.

59. (Previously Presented) The system of claim 54, wherein the at least one specification which is an encapsulation is a part of an image set for remote imaging, which henceforth renders the image set itself to be a self-describing encapsulation, encapsulating parameters resolved at execution time.

60. (Currently Amended) A system for specifying remote booting tasks in a preboot execution environment, comprising:

a computer readable medium comprising:

a language agent with a preboot execution specification generator; and

a definition for at least one specification for performing at least one task for remote booting in a preboot execution environment,

wherein the at least one specification is an encapsulation, encapsulating parameters to be resolved at a second system at execution time by a preboot execution language interpreter located at the second system, and

wherein the language agent is configured to generate one or more encapsulations and to interpret one or more encapsulations.

61. (Previously Presented) The system of claim 60, wherein the at least one specification is generated from the definition by the language agent with a preboot execution specification generator.

62. (Original) The system of claim 60, wherein the preboot execution specification generator is an object-oriented language code generator.

63. (Cancelled).

64. (Previously Presented) The system of claim 60, wherein the encapsulated parameters are parametric behaviors as well as parametric data.

65. (Previously Presented) The system of claim 60, wherein the at least one specification which is an encapsulation is a part of an image set for remote booting, which henceforth renders the image set itself to be a self-describing encapsulation, encapsulating parameters resolved at execution time.

66. (Currently Amended) A method for specifying computing tasks in a preboot execution environment, comprising ~~the steps of~~:

providing at a first system a language agent with a preboot execution specification generator;

providing at the first system at least one definition for at least one computing task in a preboot execution environment; ~~and~~

generating one or more encapsulations by the language agent; and

interpreting one or more encapsulations by the language agent

~~generating at the first system a preboot execution specification from the at least one definition utilizing the language agent with a preboot execution specification generator,~~

~~wherein the preboot execution specification is an encapsulation, encapsulating parameters to be resolved at a second system at execution time by a preboot execution language interpreter located at the second system~~.

67. (Original) The method of claim 66, wherein the preboot execution specification generator is an object-oriented language code generator.

68. (Cancelled).

69. (Currently Amended) The method of claim [[66]]112, wherein the encapsulated parameters are parametric behaviors as well as parametric data.

70. (Currently Amended) A method for specifying computing tasks for image installation in a preboot execution environment, comprising ~~the steps of~~:

providing at a first system a language agent with a preboot execution specification generator;

providing at the first system at least one definition for at least one computing task for image installation in a preboot execution environment; ~~and~~

generating one or more encapsulations by the language agent; and

interpreting one or more encapsulations by the language agent

~~generating at the first system a preboot execution specification from the at least one definition utilizing the language agent with a preboot execution specification generator,~~

~~wherein the preboot execution specification is an encapsulation, encapsulating parameters to be resolved at a second system at execution time by a preboot execution language interpreter located at the second system~~.

71. (Original) The method of claim 70, wherein the preboot execution specification generator is an object-oriented language code generator.

72. (Cancelled).

73. (Currently Amended) The method of claim [[70]]113, wherein the encapsulated parameters are parametric behaviors as well as parametric data.

74. (Currently Amended) The method of claim [[70]]113, wherein the preboot execution specification, which is an encapsulation, is a part of an image set for image installation, which henceforth renders the image set itself to be a self-describing encapsulation, encapsulating parameters resolved at execution time.

75. (Currently Amended) A method for specifying remote imaging in a preboot execution environment, comprising ~~the steps of~~:

providing at a first system a language agent with a preboot execution specification generator;

providing at the first system at least one definition for at least one computing task for remote imaging in a preboot execution environment; ~~and~~

generating one or more encapsulations by the language agent; and

interpreting one or more encapsulations by the language agent

~~generating at the first system a preboot execution specification from the at least one definition utilizing the language agent with a preboot execution specification generator,~~

~~wherein the preboot execution specification is an encapsulation, encapsulating parameters to be resolved at a second system at execution time by a preboot execution language interpreter located at the second system~~.

76. (Original) The method of claim 75, wherein the preboot execution specification generator is an object-oriented language code generator.

77. (Cancelled).

78. (Previously Presented) The method of claim 75, wherein the encapsulated parameters are parametric behaviors as well as parametric data.

79. (Previously Presented) The method of claim 75, wherein the preboot execution specification, which is an encapsulation, is itself a part of an image set for remote imaging, which henceforth renders the image set itself to be a self-describing encapsulation.

80. (Currently Amended) A method for specifying remote booting operations in a preboot execution environment, comprising ~~the steps of~~:

providing at a first system a language agent with a preboot execution specification generator;

providing at the first system at least one definition for at least one computing task for remote booting in a preboot execution environment; ~~and~~

generating one or more encapsulations by the language agent; and

interpreting one or more encapsulations by the language agent

~~generating at the first system a preboot execution specification from the at least one definition utilizing the language agent with a preboot execution specification generator,~~

~~wherein the preboot execution specification is an encapsulation, encapsulating parameters to be resolved at a second system at execution time by a preboot execution language interpreter located at the second system~~.

81. (Original) The method of claim 80, wherein the preboot execution specification generator is an object-oriented language code generator.

82. (Cancelled).

83. (Currently Amended) The method of claim [[80]]115, wherein the encapsulated parameters are parametric behaviors as well as parametric data.

84. (Currently Amended) The method of claim [[80]]115, wherein the preboot execution specification, which is an encapsulation, is itself a part of an image set for remote booting, which henceforth renders the image set itself to be a self-describing encapsulation.

85. (Currently Amended) A system for encapsulated platform imaging, comprising:

a computer readable medium comprising:

a language agent with an encapsulated language interpreter for executing an encapsulation, the language agent from a second system, the encapsulation from the second system,

wherein the encapsulation contains all instructions and data necessary to install an operating system onto a computing device, ~~and~~

wherein the encapsulation encapsulates parameters to be resolved at the system at execution time by the encapsulated language interpreter from the second system, and

wherein the language agent is configured to generate one or more encapsulations and to interpret one or more encapsulations.

86. (Original) The system of claim 85, further comprising:

a logical connection, wherein the language agent with an encapsulated language interpreter and the encapsulation is provided over the logical connection.

87. (Original) The system of claim 86, wherein the logical connection is a computer readable medium.

88. (Original) The system of claim 87, further comprising:

a bootable interface on the computer readable medium.

89. (Original) The system of claim 86, wherein the logical connection is a network connection.

90. (Original) The system of claim 89, further comprising:

a bootable interface on the network connection.

91. (Original) The system of claim 90, wherein the bootable interface on the network connection is a Preboot Execution Environment (PXE) implementation.

92. (Currently Amended) A method for encapsulated platform imaging, comprising ~~the steps of~~:

receiving by a computing device from another system an encapsulation which contains all instructions and data necessary to install an operating system onto the computing device;

receiving by the computing device from the another system a language agent with an encapsulated language interpreter for executing the encapsulation; ~~and~~

executing by the language agent at the computing device the encapsulation to install the operating system onto the computing device[[,]];

generating one or more encapsulations by the language agent; and

interpreting one or more encapsulations by the language agent,

wherein the encapsulation encapsulates parameters resolved by the language agent at the computing device at execution time, the language agent located at the computing device.

93. (Original) The method of claim 92, wherein the encapsulation and the language interpreter are provided over a logical connection.

94. (Original) The method of claim 93, wherein the logical connection is a computer readable medium.

95. (Currently Amended) The method of claim 94, further comprising ~~the steps of~~:

providing a bootable interface on the computer readable medium.

96. (Currently Amended) The method of claim 95, further comprising ~~the steps of~~:

booting from the computer readable medium;

loading the language agent with an encapsulated language interpreter for executing the encapsulation from the computer readable medium; and

loading the encapsulation from the computer readable medium before executing the encapsulation.

97. (Original) The method of claim 93, wherein the logical connection is a network connection.

98. (Currently Amended) The method of claim 97, further comprising ~~the steps of~~:

providing a bootable interface on the network connection.

99. (Currently Amended) The method of claim 98, further comprising ~~the steps of~~:

booting over the network connection;

loading the language agent with an encapsulated language interpreter for executing the encapsulation over the network connection; and

loading the encapsulation over the network connection before executing the encapsulation.

100. (Original) The method of claim 98, wherein the bootable interface on the network connection is a Preboot Execution Environment (PXE) implementation.

101. (Currently Amended) A system for encapsulated platform imaging, comprising:

a computer readable medium comprising:

a language agent with an encapsulation generator for defining and creating an encapsulation,

wherein the encapsulation contains all instructions and data necessary to install an operating system onto a computing device,

wherein the encapsulation encapsulates parameters to be resolved at a second system at execution time by a language interpreter located at the second system, and

wherein the language agent is configured to generate one or more encapsulations and to interpret one or more encapsulations.

102. (Currently Amended) A method for encapsulated platform imaging, comprising ~~the steps of~~:

providing at a first system a language agent with an encapsulation generator;

providing at the first system a definition for an encapsulation; ~~and~~

generating at the first system from the definition an encapsulation containing all instructions and data necessary to install an operating system onto a computing device by utilizing the language agent with an encapsulation generator[[,]];

generating one or more encapsulations by the language agent; and

interpreting one or more encapsulations by the language agent,

wherein the encapsulation encapsulates parameters to be resolved at a second system at execution time by a language interpreter located at the second system.

103. (Currently Amended) A system for executing computing tasks in a preboot execution environment, comprising ~~the steps of~~:

means for receiving by the system a language agent with a preboot execution language interpreter, the language agent originated from a second system;

means for receiving by the system at least one specification for performing at least one computing task in the preboot execution environment, the at least one specification originated from the second system;

~~means for interpreting by the language agent the at least one specification for performing at least one computing task in the preboot execution environment;~~ and

means for performing the at least one computing task specified,

wherein the at least one specification is an encapsulation, encapsulating parameters resolved at the system at execution time by the preboot execution language interpreter from the second system,

wherein the language agent is configured to interpret the at least one specification for performing at least one computing task in the preboot execution environment, and

wherein the language agent is configured to generate one or more encapsulations and to interpret one or more encapsulations.

104. (Currently Amended) A system for specifying computing tasks in a preboot execution environment, comprising:

means for providing at the system a language agent with a preboot execution specification generator; and

means for providing at the system at least one definition for at least one computing task in a preboot execution environment; ~~and~~

~~means for generating at the system a preboot execution specification from the at least one definition utilizing the language agent with a preboot execution specification generator,~~

~~wherein the preboot execution specification is an encapsulation, encapsulating parameters to be resolved at a second system at execution time by a preboot execution language interpreter located at the second system,~~

wherein the language agent is configured to generate one or more encapsulations and to interpret one or more encapsulations.

.

105. (Previously Presented) The system of claim 1, wherein the at least one computing task includes copying or applying an endpoint to an instance.

106. (Previously Presented) The system of claim 1, wherein the at least one computing task includes a de-referencing operation.

107. (Previously Presented) The system of claim 1, wherein the at least one computing task includes initiating or terminating encapsulation of archives.

108. (Previously Presented) The system of claim 1, wherein the at least one computing task includes referencing archives from a single archive.

109. (Previously Presented) The system of claim 1, wherein the language agent is configured to generate encapsulations and to interpret the encapsulations.

110. (Previously Presented) The system of claim 1, wherein while executing an encapsulation, the language agent is configured to generate another encapsulation that is to be executed at a later time.

111. (Previously Presented) The system of claim 101, wherein no programs or codes are preserved separate from the encapsulation.

112. (New) The method of claim 66, further comprising:

generating at the first system a preboot execution specification from the at least one definition utilizing the language agent,

wherein the preboot execution specification is an encapsulation, encapsulating parameters to be resolved at a second system at execution time by a preboot execution language interpreter located at the second system.

113. (New) The method of claim 70, further comprising:

generating at the first system a preboot execution specification from the at least one definition utilizing the language agent,

wherein the preboot execution specification is an encapsulation, encapsulating parameters to be resolved at a second system at execution time by a preboot execution language interpreter located at the second system.

114. (New) The method of claim 75, further comprising:

generating at the first system a preboot execution specification from the at least one definition utilizing the language agent,

wherein the preboot execution specification is an encapsulation, encapsulating parameters to be resolved at a second system at execution time by a preboot execution language interpreter located at the second system

115. (New) The method of claim 80, further comprising:

generating at the first system a preboot execution specification from the at least one definition utilizing the language agent,

wherein the preboot execution specification is an encapsulation, encapsulating parameters to be resolved at a second system at execution time by a preboot execution language interpreter located at the second system.

116. (New) The system of claim 104, further comprising:

means for generating at the system a preboot execution specification from the at least one definition utilizing the language agent,

wherein the preboot execution specification is an encapsulation, encapsulating parameters to be resolved at a second system at execution time by a preboot execution language interpreter located at the second system.